UNIVERSIDAD AUTÓNOMA DE MADRID
ESCUELA POLITÉCNICA SUPERIOR

TRABAJO FIN DE GRADO

MONITORIZACIÓN DE PATRONES BIOMÉTRICOS PARA APLICACIONES WEB

Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación

Elena Luna García
Tutor: Dr. Aythami Morales Moreno
Ponente: Dr. Julián Fiérrez Aguilar

Junio 2015

MONITORIZACIÓN DE PATRONES BIOMÉTRICOS PARA APLICACIONES WEB

Elena Luna García
Tutor: Dr. Aythami Morales Moreno

Biometric Recognition Group - ATVS
Departamento de Tecnología Electrónica y de las Comunicaciones
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Junio 2015

Resumen.

El desarrollo tecnológico de los últimos años ha contribuido al acceso y almacenamiento masivo de información digital. Este creciente desarrollo junto con la proliferación de los servicios web, conllevan la necesidad de sistemas de gestión de indentidades a gran escala. La seguridad cibernética es un asunto de gran importancia que concierne a gobiernos, empresas y usuarios. Frente a las numerosas desventajas que presentan los actuales sistemas de autenticación basados en contraseñas, como la pérdida, el robo o el hackeo, los sistemas de autenticación basados en reconocimiento biométrico se presentan como potenciales sustitutos.

La dinámica de tecleo se ha convertido en un área de investigación activa debido a su bajo coste y su fácil integración en los dispositivos. Al tratarse de un rasgo biométrico de comportamiento, sus datos y resultados presentan una gran variabilidad. La normalización de resultados ha demostrado ser una técnica útil para mitigar estos efectos.

En este trabajo se estudia la utilidad de la normalización de resultados en un sistema basado en dinámica de tecleo. Se realiza una evaluación de diferentes algoritmos de clasificación en dos bases de datos de características diferentes, una de desarrollo propio y otra de disponibilidad pública, estudiando el impacto de la normalización en cada una de ellas y el rendimiento de cada algoritmo.

Finalmente se presentan posibles objetivos para mejorar los resultados obtenidos y líneas de trabajo futuras.

Palabras clave

Biometría, reconocimiento biométrico, reconocimiento de patrones, autenticación, dinámica de tecleo, base de datos, características, clasificación, normalización.

Abstract.

Technological development in recent years has contributed to the access and massive storage of digital information. This increasing development along with the proliferation of Web services, involve the need of management systems of large-scale identities. Cyber Security is a matter of great importance that concerns governments, companies and users. Because of the many disadvantages of the current password-based authentication systems, such as loss, theft or hacking, systems based on biometric authentication are potencial substitutes of them.

Keystroke dynamics has become an active area of research because of its low cost and easy integration into devices. Due to being a behavioural biometric feature, data and results are highly variable. Score normalization has proved a useful technique to mitigate this effects.

This work studies the usefulness of the score normalization in a system based on keystroke dynamics. It assesses differents algorithms in two different databases, one of own development and other public database. It studies the normalization impact in each database and evaluates the performance of each algorithm.

Finally, potencial targets, to improve the results, and future research are presented.

Key words

Biometrics, biometric recognition, pattern recognition, authentication, keystroke dynamics, database, features, classification, score normalization.

Agradecimientos.

En primer lugar, quisiera dar las gracias a mi tutor, Aythami Morales, por darme la oportunidad de realizar este trabajo, por guiarme en todo el proceso y ayudarme en todo lo que he necesitado en todo momento.
Agradecer a Mario su colaboración en este trabajo, sin la cual no hubiera podido llevarse a cabo.
Quiero agradecer también a todas las personas que han colaborado en la creación de la base de datos su paciencia y tiempo invertido.
A todas las personas que han estado a mi lado a lo largo de estos cuatro años, que han hecho de ellos unos años para recordar. A los que han estado desde el principio, a los que he conocido más adelante y a los que me acompañan en esta última etapa.
A Víctor, por estar a mi lado y darme ánimos y apoyo siempre.
Y por supuesto a mi familia, a mis padres y hermana, que han vivido todo esto conmigo, por confiar en mí, apoyarme y hacer de mí quien soy ahora. A mis abuelos, les dedico este trabajo, por su apoyo y preocupación.
Para todos vosotros, gracias.

Glosario

- **RNA**: Red Neuronal Artificial.
- **KNN**: K Nearest Neighbors.
- **SVM**: Support Vector Machine.
- **CMU**: Carneige Mellon University.
- **PA**: Pennsylvania.
- **EEUU**: Estados Unidos.
- **ESTA**: Electronic System for Travel Authorization.
- **H**: Hold time.
- **Latencia AP**: Latencia Alzado-Presión.
- **Latencia PA**: Latencia Presión-Alzado.
- **Latencia PP**: Latencia Presión-Presión.
- **Latencia AA**: Latencia Alzado-Alzado.
- **DNI**: Documento Nacional de Identidad.
- **FAR**: False Acceptance Rate.
- **FRR**: False Rejection Rate.
- **ROC**: Receiver Operating Characteristic.
- **EER**: Equal Error Rate.
- **TIC:** Target-Impostor Centric.
- **TC:** Target Centric.
- **LOOCV**: Leave-One-Out Cross-Validation.

Índice general

Índice de figuras

Índice de tablas

Capítulo 1

Introducción.

1.1. Reconocimiento de patrones.

El reconocimiento de patrones es la disciplina científica que tiene por objetivo la categorización de objetos mediante la extracción automática de información y su clasificación. Dependiendo de la aplicación, estos objetos pueden ser imágenes, señales o cualquier otro tipo de medida que sea necesario clasificar. El término *patrones* hace referencia a estos objetos. En función de las características extraídas del análisis se define el patrón, y todos los patrones que presentan características similares se agrupan en clases. El principal objetivo del reconocimiento de patrones es llevar a cabo la asignación de clases de manera automática.
Un sistema completo de reconocimiento de patrones, véase la figura 1.1, está formado por un sensor, un sistema de extracción de características y un clasificador.

El sensor es el dispositivo encargado de la adquisición de los datos, y la calidad de estos depende de las características tecnológicas del mismo. El sistema de extracción de características puede estar precedido por un sistema de preprocesamiento de los datos adquiridos por el sensor para mejorar la calidad de estos, por ejemplo, un filtrado que elimine posible ruido de los datos. El módulo extractor de características transforma los datos en valores cuantificables con el objetivo de extraer un conjunto

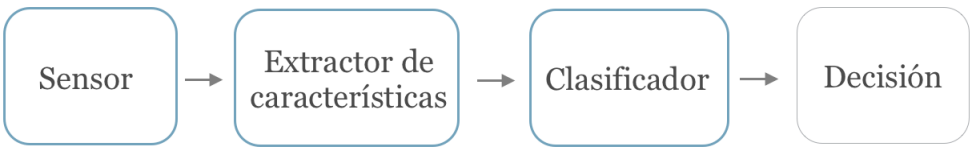

Figura 1.1: Sistema completo de reconocimiento de patrones.

de características discriminatorias que permitan representar un modelo o clase. El clasificador emplea las características obtenidas para asignar al objeto una categoría u otra.
El reconocimiento de patrones es el centro de numerosas áreas de aplicación, incluyendo el análisis de imagen, el reconocimiento de voz y audio, la biometría, la ingeniería biomédica, la minería de datos e investigaciones geológicas y geofísicas. A pesar de sus notables diferencias, todas estas áreas comparten, en gran medida, un conjunto de técnicas que pueden ser empleadas en la extracción, a partir de los datos disponibles, de la información relativa a las categorías de datos, como pueden ser patrones ocultos y tendencias.

1.2. Reconocimiento biométrico.

El reconocimiento de patrones es la principal base teórica de la biometría. La biometría es definida, por el diccionario de la Real Academia de la Lengua Española, como el estudio mensurativo o estadístico de los fenómenos o procesos biológicos. Sin embargo, el reconocimiento biométrico se puede definir como la disciplina que se encarga de establecer la identidad de un individuo en base a las características físicas, químicas o atributos de comportamiento de la persona [3].
Existen evidencias de que los humanos han usado sus características corporales tales como la cara, la voz y huellas dactilares y palmares durante años para identificarse mutuamente. A mediados del siglo XIX, Alphonse Bertillon, jefe de la división de indentificación criminal del Departamento de Policía de París, desarrolló y puso en práctica la idea de emplear medidas corporales para la identificación criminal. Del mismo modo que esta idea fue ganando popularidad, fue eclipsada por un descubrimiento mucho más significativo y práctico sobre la caracterización de las huellas dactilares humanas a finales de siglo [4]. Aunque la biometría emergió de su extenso uso en el ámbito legal para la identificación de criminales, hoy en día se emplea cada vez más en el reconocimiento de personas en numerosas aplicaciones civiles de uso cotidiano.
La mayor ventaja de la aplicación de la biometría para la identificación de las personas radica en que la autenticación se basa en la propia persona, no en algo que sepa o que lleve consigo. Al contrario de los métodos clásicos como pueden ser las llaves, tarjetas, contraseñas o códigos, las características biométricas no son fáciles de modificar, perder, ceder a otra persona u olvidar.
Existen numerosos rasgos biométricos que pueden ser usados para diferentes aplicaciones. En la figura 1.2 se muestran algunos ejemplos. Los rasgos biométricos se

clasifican en dos grupos: los rasgos físicos, como son la voz, las huellas dactilares y palmares, el iris, la retina y la geometría facial, entre otros, y los rasgos referentes al comportamiento, como la escritura, la firma o la dinámica de tecleo, rasgo en el cual se centra este trabajo. Sin embargo, la voz y la escritura son rasgos que dependen, en cierto modo, tanto de las características físicas como del comportamiento, ya que son algo que se aprende, por tanto, se pueden clasificar en ambos grupos indistintamente.

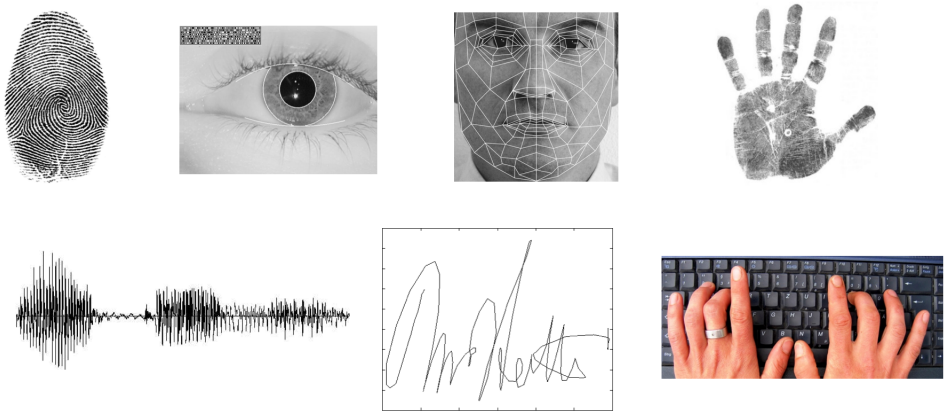

Figura 1.2: Ejemplos de rasgos biométricos.

Teóricamente, cualquier característica humana puede ser considerada y aplicada como un rasgo biométrico, pero se deben considerar siete factores que determinan la idoneidad de un rasgo físico o de comportamiento a la hora de ser usado en una aplicación biométrica [5]:

- **Universalidad**: todos los individuos que accedan a la aplicación deben poseer el rasgo en cuestión.
- **Unicidad**: el rasgo debe ser suficientemente diferente entre los individuos, es decir, debe ser discriminatorio.
- **Perdurabilidad**: el rasgo debe ser suficientemente invariable a lo largo del tiempo.
- **Mensurabilidad**: debe ser posible adquirir y digitalizar el rasgo biométrico a partir del uso de dispositivos adecuados que no ocasionen molestias al individuo.
- **Rendimiento**: a parte de la precisión del reconocimiento, los recursos computacionales requeridos y la velocidad del sistema, el sistema también debe cumplir las restricciones impuestas por la aplicación.

- **Aceptabilidad**: los individuos deben estar dispuestos a presentar su rasgo biométrico al sistema.
- **Elusión**: referente a la facilidad con la que el rasgo puede ser imitado usando mecanismos para ello, como por ejemplo, fotografías o dedos falsos para el caso de rasgos físicos y la imitación, en caso de rasgos de comportamiento.

Ningún rasgo biométrico supera al resto en todos los campos ni cumple satisfactoriamente todos los requisitos, cada uno presenta ventajas y desventajas. La aplicación de uno u otro depende, básicamente, de los requisitos que imponga la aplicación a desarrollar.
La primera etapa de todo sistema biométrico es el **registro** de usuarios, se muestra su esquema de funcionamiento en la figura 1.3. Durante esta etapa, el sensor adquiere los datos biométricos a través de la interfaz de usuario. Los datos pueden ser opcionalmente preprocesados para mejorar su calidad y a continuación se procede a la extracción de las características identificativas que serán almacenadas, junto a la identidad del usuario, en la base de datos. Las características almacenadas pueden servir como referencia o ser empleadas para entrenar un modelo probabilístico de la identidad a la que pertenecen, por ello, esta etapa se conoce también como **fase de entrenamiento.**

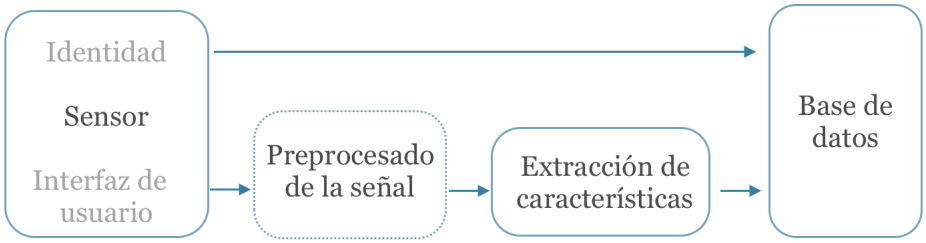

Figura 1.3: Esquema de la etapa de registro de un sistema biométrico. Figura adaptada de [1].

Una vez registrados los usuarios, el sistema biométrico puede operar en dos modos de funcionamiento: indentificación y verificación, en función de la aplicación que se esté llevando a cabo [4].
En el modo de **identificación**, cuyo esquema se muestra en la figura 1.4, el sensor adquiere las características del usuario que se quiere identificar y realiza una comparación con todos los datos de la base de datos. A la salida del comparador se obtiene la identidad del usuario, en caso de que se encuentre en la base de datos, o una lista ordenada de candidatos.

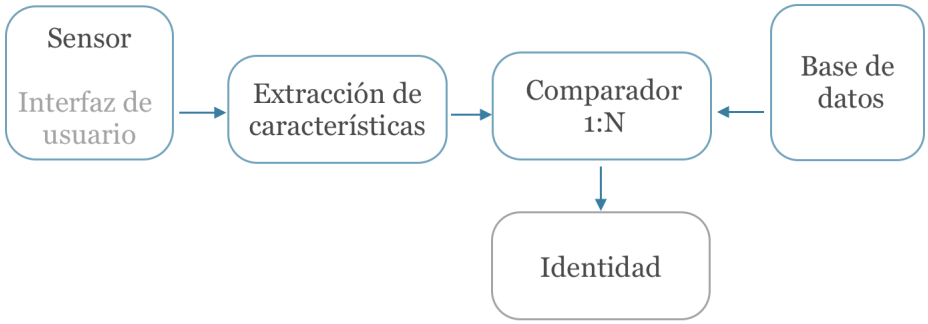

Figura 1.4: Esquema de la etapa de identificación de un sistema biométrico. Figura adaptada de [1].

En el modo de **verificación**, ver figura 1.5, el usuario proporciona su identidad y el sensor, mediante la interfaz de usuario, adquiere el rasgo biométrico del que posteriormente se extraen las características y se comparan con las características de dicho usuario almacenadas en la base de datos. A la salida se obtiene si la verificación ha resultado correcta o incorrecta.

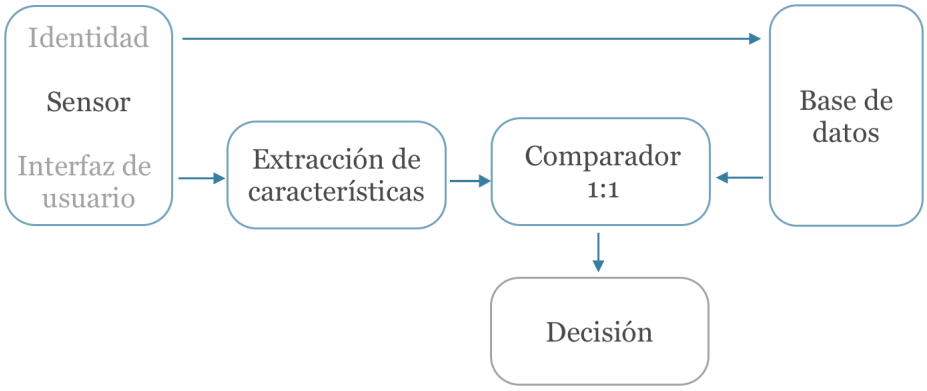

Figura 1.5: Esquema de la etapa de verificación de un sistema biométrico. Figura adaptada de [1].

1.3. Reconocimiento basado en la dinámica de tecleo.

La dinámica de tecleo es un rasgo biométrico de comportamiento que consiste en analizar la cadencia de tecleo de los usuarios. Se pueden esperar grandes variaciones intra-clase en los patrones de tecleo de una persona, es decir, un mismo individuo

puede realizar diferentes versiones genuinas de un mismo patrón de tecleo, debido a cambios emocionales, posición del usuario frente al teclado, tipo de teclado que se emplee, etc. Este rasgo no se espera que sea único para cada individuo, pero sí se puede pensar que ofrece información lo suficientemente discriminatoria para permitir la verificación de identidad. Cuando una persona teclea, la latencia entre pulsaciones sucesivas [6], la duracion de las mismas, la posición de los dedos, la presión aplicada a las teclas [7], e incluso el sonido producido al teclear [8] son parámetros que pueden ser usados para construir una firma única para cada usuario [9]. Además, capturar la dinámica de tecleo no es un método intrusivo, el tecleo puede ser monitorizado discretamente, sin que la persona tenga constancia de ello, lo que lo hace fácilmente integrable en cualquier plataforma o servicio relacionado con la interacción con un teclado. Esto permite una verificación continua de la identidad del usuario con un coste mínimo, ya que el único hardware que se requiere es un teclado.

La técnica de verificación basada en la dinámica de tecleo puede ser considerada tanto estática como continua. Los enfoques de verificación estáticos analizan los patrones de tecleo sólo en momentos específicos, por ejemplo, durante una secuencia de inicio de sesión. Este enfoque proporciona una verificación de usuarios más robusta que una simple contraseña, pero no proporciona seguridad continua, es decir, no puede detectar una sustitución del usuario después de la verificación inicial. La verificación continua, por el contrario, monitoriza el comportamiento del usuario durante todo el transcurso de la sesión.

El uso de la dinámica de tecleo para identificación y verificación tiene una larga historia cuyos inicios datan de la década de 1970 [10, 11]. Ya en 1980 Gaines et al. [12] publicaron un amplio y exhaustivo artículo de investigación y análisis sobre la autenticación basada en la dinámica de tecleo, el cual fue especialmente revelador. Sus investigaciones muestran que los inicios de este campo tienen su origen en el tecleo de código morse en los telégrafos, se observó que cada operador tenía una manera particular de teclear los mensajes, con su propio ritmo. Esto resultó muy útil durante la Segunda Guerra Mundial, lo que les permitía identificar al emisor del mensaje. Empleaban una metodología conocida como “Fist of the Sender”, traducido como “El Puño del Emisor”, que les permitía identificar al emisor a partir del ritmo y síncopa de las teclas de telégrafo. Mientras Gaines et al. llegaron a la conclusión de que un sistema de este tipo podría resultar eficaz para la autenticación, reconocieron que los resultados se basaban solamente en una pequeña muestra, se trataba de datos de siete mecanógrafos que habían tecleado tres secciones diferentes de un texto, con cuatro meses de diferencia entre cada sesión. Sin embargo, a pesar del reducido tamaño de la muestra, los investigadores fueron capaces de observar diferencias entre los distintos

estilos de tecleo.
El estudio de Gaines et al. [12] popularizó el uso de datos dígrafos, es decir, datos asociados a dos letras tecleadas sucesivamente, este método allanó el camino a muchos grupos de análisis de dinámica de tecleo posteriores para forjar un primer recorrido en este campo, el cual mantiene aún su popularidad.
Partiendo de la apreciación de Gaines et al., otros investigadores han continuado investigaciones sobre este campo persiguiendo, sobre todo, la reducción de las tasas de error. En la sección 2.4 se describen las tasas que miden el rendimiento.
Desde entonces, se han desarrollado y publicado numerosos algoritmos de comparación y de aprendizaje automático para abordar el problema [6, 9, 13, 14]. Los algoritmos más populares se pueden clasificar en cuatro categorías: métodos estadísticos basados en medidas de distancia, redes neuronales, métodos estadísticos de aprendizaje automático y métodos heurísticos combinación de varios algoritmos [15]. La primera categoría emplea estadísticas de primer y segundo orden de la característica y las distancias métricas aplicadas. Se han estudiado numerosas distancias posibles como la distancia Euclídea [9], la distancia Mahalanobis [16], la distancia Manhattan [6] y combinaciones de ellas. Las redes neuronales artificiales (RNA) son modelos que intentan emular el comportamiento neuronal humano. Son sistemas de aprendizaje adaptativo autoorganizado capaces de aprender modelos no lineales de datos, sin embargo, a menudo presentan baja velocidad en el proceso de entrenamiento del modelo, selección manual del modelo de la arquitectura y los parámetros de ajuste, y pobres capacidades de generalización [15]. Los algoritmos de aprendizaje automático abarcan desde los simples clasificadores K-vecinos más cercanos (KNN) [16], hasta los clasificadores Bayesianos [9] y las Máquinas de Soporte Vectorial (SVM) [17]. Comparadas con las redes neuronales artificiales, las Máquinas de Soporte Vectorial requieren el ajuste de menos parámetros y pueden resultar mucho más eficientes tanto en la fase de entrenamiento como en la de verificación [15].
A continuación, se muestra una evaluación de múltiples algoritmos probados en la base de datos CMU (Universidad Carnegie Mellon, PA EEUU) [2]. En la tabla 1.1 se muestran los resultados de los 14 mejores clasificadores, ordenados de mejor a peor rendimiento.

1.4. Normalización de resultados de clasificador.

La normalización de los datos es un paso fundamental en cualquier sistema de reconocimiento de patrones. Dependiendo de la etapa en la que se lleve a cabo podemos distinguir diferentes tipos de normalización:

Table 1.1: Evaluación de clasificadores con la base de datos CMU. Se muestra la tasa de igual error media (con la desviación estándar entre paréntesis). Datos obtenidos de [2].

	Clasificador	EER (stdev)
1	Manhattan (scaled)	0.096 (0.069)
2	Nearest Neighbor (Mahalanobis)	0.100 (0.064)
3	Outlier Count (z-score)	0.102 (0.077)
4	SVM (one-class)	0.102 (0.065)
5	Mahalanobis	0.110 (0.065)
6	Mahalanobis (normed)	0.110 (0.065)
7	Manhattan (filter)	0.136 (0.083)
8	Manhattan	0.153 (0.092)
9	Neural Network (auto-assoc)	0.161 (0.080)
10	Euclidean	0.171 (0.095)
11	Euclidean (normed)	0.215 (0.119)
12	Fuzzy Logic	0.221 (0.105)
13	k Means	0.372 (0.139)
14	Neural Network (standard)	0.828 (0.148)

- **Normalización de los datos de entrada:** Como hemos visto al principio del capítulo 1, todo sistema biométrico puede contar con un bloque de preprocesamiento de los datos obtenidos por el sensor, previo al extractor de características. El objetivo de este preprocesado es la mejora de la calidad de los datos, y una de las tareas que se puede llevar a cabo para ello es la normalización de los datos de entrada. En este trabajo, los datos capturados son marcas de tiempo del instante temporal en el que una tecla es presionada. A estos datos se les aplica una normalización sobre el eje temporal. Las marcas de tiempo obtenidas toman como referencia temporal inicial el instante temporal marcado por el ordenador en el que tiene lugar la adquisición, el cual es distinto para cada una. Además, este instante podía diferir de un ordenador a otro. Es necesario, entonces, aplicar una normalización que determine el instante en el que se pulsa la primera tecla como el inicial, igual para todas las adquisiciones.

- **Normalización de características**: Un ejemplo de este tipo de normalización es el llevado a cabo en el cálculo de la distancia Manhattan escalada, que se detalla en la sección 2.2.2. En [18] se propone utilizar la desviación estándar de las características para su normalización. La desviación típica o estándar es una medida de dispersión que indica cuánto se aleja una muestra de la media del conjunto. Un usuario que presenta una desviación baja indica que tiene una

cadencia de tecleo estable, mientras que si un usuario presenta una desviación alta indica lo contrario, que no tiene un ritmo estable definido. La división entre la desviación estándar tiene por objetivo ponderar las características en función de su estabilidad, es decir, según su habilidad para discriminar usuarios.

- **Normalización de resultados**: Es el tipo de normalización en la que se centra este trabajo. Los resultados a la salida de un clasificador no tienen por qué ser homogéneos, por ejemplo, un clasificador puede producir a su salida una medida de disimilitud, mientras otro puede dar una medida de similitud para los mismos datos. Además, los resultados de un clasificador no tienen por qué presentar todos la misma escala o seguir la misma distribución estadística. A todo estoy hay que añadir la alta dependencia de usuario de un rasgo de comportamiento como la dinámica de tecleo [19]. Por estas razones, la normalización de los resultados a un dominio común es una tarea esencial. La normalización de resultados se define como la modificación del rango y la escala de las distribuciones de los resultados de un clasificador, de manera que los resultados de distintos comparadores presenten un dominio común. En un buen sistema de normalización, la elección de los parámetros de rango y escala debe ser robusta y eficiente. La robustez hace referencia a la insensibilidad a la presencia de valores atípicos. La eficiencia trata cómo de cerca está la estimación obtenida de una estimación óptima, cuando las distribuciones de los datos son conocidas [20].

A diferencia de los rasgos biométricos físicos, como pueden ser la huella dactilar o el iris, los cuales presentan distribuciones de resultados similares en diferentes usuarios, la dinámica de tecleo es un rasgo biométrico de comportamiento, sección 1.3, que puede presentar innumerables variaciones entre individuos e incluso en uno mismo. Esta inherente naturaleza basada en comportamiento hace que los datos de entrada se ajusten mejor a un proceso aleatorio que a una señal determinista [21]. De esta problemática, junto a la vista anteriormente, surge la necesidad de normalizar los resultados. Existen dos tipos de normalización, la normalización de resultados dependiente del usuario genuino y la normalización dependiente del conjunto de verificación. La normalización dependiente del usuario emplea muestras genuinas e impostoras de otros usuarios que intentan imitar el comportamiento del usuario genuino, aunque se trate de resultados genuinos e impostores, ambos casos están relacionados con un mismo usuario. La normalización dependiente del conjunto de verificación, o conjunto de test, emplea las muestras de verificación y modelos de los usuarios impostores [22].

1.5. Motivación y objetivos.

La importancia del reconocimiento biométrico se ha incrementado notablemente en la sociedad actual debido a la necesidad de sistemas de gestión de identidades a gran escala, cuyo propósito es la determinación de la identidad individual en diversos contextos y aplicaciones. La proliferación de los servicios web, como la banca online, y del uso de dispositivos como las tarjetas de crédito, han remarcado la necesidad de una identidad segura. La seguridad cibernética es una asunto crítico que concierne a los gobiernos, las empresas y a todos los usuarios que hacen uso de la red. Actualmente, la autenticación basada en contraseñas es el sistema de verificación/acceso más empleado, pero la seguridad de estos sistemas depende, en gran medida, de la robustez de la contraseña. Este sistema presenta varios problemas como la pérdida, el robo o el hackeo de dichas contraseñas, por ello, los sistemas biométricos son buenos sustitutos de este sistema.
En los últimos tiempos, la dinámica de tecleo se ha convertido en un área de investigación muy activa debido a la ya mencionada importancia de la seguridad cibernética y el control de acceso a ordenadores y a la red. La problemática que puede presentar este sistema basado en comportamiento es, como hemos visto en la sección 1.4, la alta variabilidad que presenta en entornos reales. A continuación se ilustra esta problemática. En la figura 1.6 se muestran las curvas FAR y FRR para cuatro usuarios de la base de datos CMU.
Se observa que los cuatro umbrales óptimos de decisión presentan gran variabilidad, debido a la cual estimar un umbral para todos los usuarios es una tarea difícil. En la figura 1.7 se ilustra el impacto de una selección de umbral de clasificación basada en los resultados de un único usuario. En la figura 1.7 (izquierda) se observan las curvas FAR (Tasa de Falsa Aceptación, ver sección 2.4) y FRR (Tasa de Falso Rechazo, sección 2.4) de un usuario, así como el umbral óptimo calculado a partir del EER (Tasa de Igual Error, explicada en la sección 2.4). En la figura 1.7 (derecha) se observa el impacto que tendría mantener el umbral del usuario anterior, aumentando la tasa de error (FAR) de manera considerablemente un 80 %.
La normalización de resultados ya ha demostrado su utilidad frente a este problema en otros campos tales como la voz [23] o la firma manuscrita [24].
Los objetivos de este trabajo son:

- Estudio de la utilidad de la normalización de resultados en un sistema basado en la dinámica de tecleo.
- Estudio del impacto de la normalización en distintas bases de datos y la problemática de cada una.

- Estudio del impacto de la normalización en distintos clasificadores.

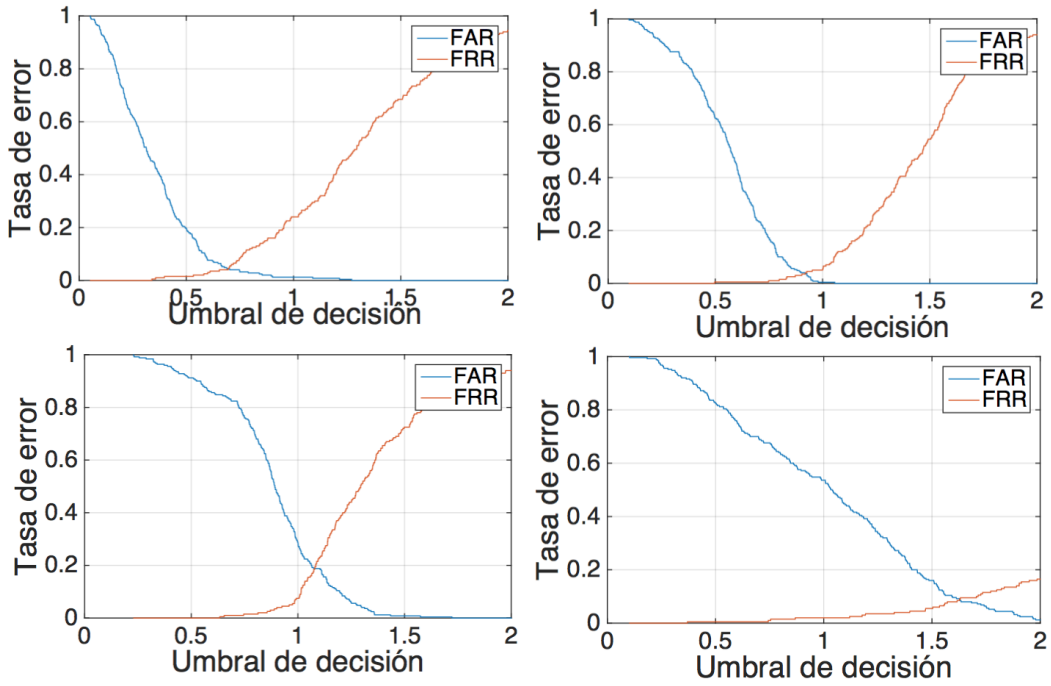

Figura 1.6: Comparativa curvas FAR y FRR y umbrales de decisión para 4 usuarios.

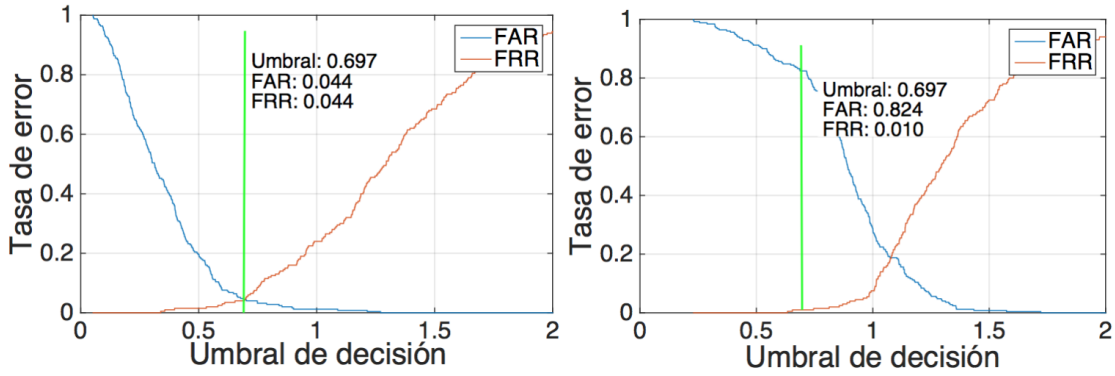

Figura 1.7: Impacto de la selección de un umbral de clasificación único basada en los resultados de un único usuario.

1.6. Estructura de la memoria.

La memoria de este trabajo se divide en los siguientes capítulos:

- **Capítulo 1: Introducción**. Introducción al reconocimiento de patrones, al reconocimiento biométrico y a la normalización de resultados, motivación y objetivos del proyecto.
- **Capítulo 2: Metodología**. Presentación de las dos bases de datos empleadas, los algoritmos de reconocimiento, los métodos de normalización estudiados y las medidas de rendimiento utilizadas.
- **Capítulo 3: Experimentos.** Presentación de los protocolos de experimentación y los resultados obtenidos para las dos bases de datos estudiadas.
- **Capítulo 4: Conclusiones y trabajo futuro**. Presentación de las conclusiones extraídas y las futuras líneas de trabajo que se plantean.

Capítulo 2

Metodología.

2.1. Bases de datos.

La disponibilidad de bases de datos y marcos experimentales es esencial para el desarrollo de las tecnologías de reconocimiento biométrico. El reconocimiento biométrico basado en la dinámica de tecleo depende, en gran manera, del usuario y la aplicación empleada para la adquisición. Las variaciones intra-clase (entre muestras del mismo usuario) e inter-clase (entre muestras de diferentes usuarios), al tratarse de un rasgo de comportamiento, se ven afectadas por factores referentes al usuario y a la aplicación. En la tabla 2.1 se muestran algunas de las bases de datos de dinámica de tecleo más populares.

Table 2.1: Bases de datos públicas disponibles para el reconocimiento basado en la dinámica de tecleo.

Base de datos	**Usuarios**	**Muestras**	**Sesiones**	**Passwords**
CMU [2]	51	400	8	*tie5Roanl*
MIMOS [14]	100	10		*try4-mbs*
Clarkson [25]	39	20	1	*yesnomaybe*, *bahaNe312*, *ballzoneca*
GREYC [26]	133	7555	5	*greyc laboratory*
BeiHang [27]	117	4-16	1	Distinta para cada usuario

La mayoría de las bases de datos públicas se basan en una única contraseña, o *password,* para todos los usuarios: *tie5Roanl* (CMU [2]), *try4-mbs* (MIMOS [14]), etc. Sin embargo, aunque en la realidad los servicios de autenticación se basan en contraseñas personales únicas, ya sean creadas por el usuario o generadas por el sistema, las bases de datos basadas en este escenario real, con una contraseña única y personal, son

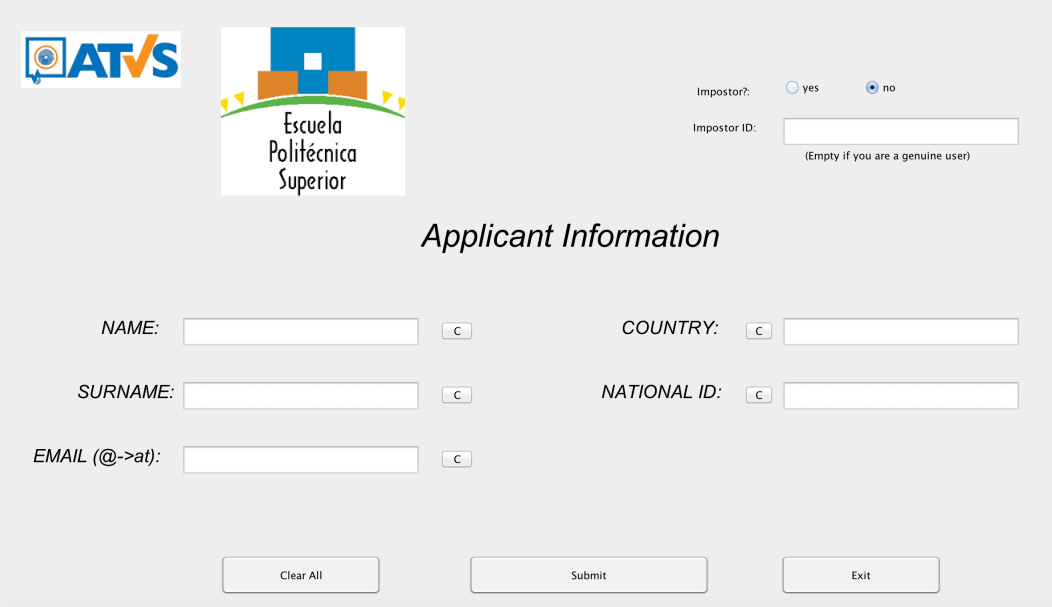

Figura 2.1: Interfaz de la aplicación de adquisición.

escasas.

2.1.1. Base de datos de desarrollo propio.

En la primera fase de este trabajo se participó en la adquisición de una base de datos propia. Para la adquisición de los datos se ha empleado una aplicación java, basada en la librería *java.awt.event.KeyListener* para la detección de las pulsaciones y liberaciones de las teclas, que consta de una interfaz gráfica de usuario a través de la cual los usuarios introducen los datos requeridos [28]. En la figura 2.1 se muestra la interfaz de la aplicación. La aplicación está inspirada en el formulario electrónico ESTA (*Electronic System for Travel Authorization*), usado por Estados Unidos para mejorar la seguridad referente a los extranjeros que viajan al país. Nuestra base de datos está formada por 64 usuarios, en su mayoría españoles e italianos. La novedad de esta base de datos es que en lugar de solicitar la introducción de contraseñas o *passwords* generados para la autenticación del usuario, se solicita información personal (nombre, apellidos, correo, nacionalidad y DNI) para analizar la dinámica de tecleo de este tipo de datos. Se espera que al ser datos muy comunes, los usuarios tengan una elevada práctica en su tecleo y eso produzca patrones más estables. Además, los datos personales no requieren ser memorizados, lo que mejoraría la usabilidad de este tipo de sistemas, basada habitualmente en complejas cadenas de caracteres.
En una primera sesión genuina los usuarios realizaron 6 adquisiciones en las cuales

debían introducir sus datos personales: nombre, apellidos, correo, nacionalidad y DNI. Tras un período de tiempo, de mínimo 24 horas, se llevó a cabo la segunda sesión genuina, en la cual cada usuario realizó otras 6 adquisiciones introduciendo de nuevo los mismos datos. Tras finalizar las adquisiciones genuinas, sin necesidad de que transcurriera un período de tiempo determinado, se realizó la sesión impostora, en la cual se proporcionó a cada usuario los datos personales de otros 3 usuarios. En esta sesión, cada usuario tuvo que realizar 4 adquisiciones de cada usuario al que suplantaba. De esta manera, una vez finalizadas todas las adquisiciones, cada usuario tiene un total de 12 adquisiciones genuinas y 12 adquisiciones impostoras. Todas las adquisiciones se realizaron bajo supervisión y si ocurría algún error en la adquisición, se repetía de nuevo (por ejemplo, algún fallo en algún carácter).
La aplicación origina a su salida un fichero de texto en el que aparecen los instantes de tiempo absolutos en que cada tecla es presionada y liberada, ver figura 2.2. Posteriormente estos ficheros son procesados para la extracción de características, ver sección 2.2.1.

```
NAME[elena]:
[pressedE] 1417606554665 [releasedE] 1417606554764
[pressedL] 1417606554795 [releasedL] 1417606554862
[pressedE] 1417606554894 [releasedE] 1417606555024
[pressedN] 1417606555027 [releasedN] 1417606555122
[pressedA] 1417606555185 [releasedA] 1417606555284

SURNAME[luna]:
[pressedL] 1417606556773 [releasedL] 1417606556840
[pressedU] 1417606556966 [releasedU] 1417606557033
[pressedN] 1417606557128 [releasedN] 1417606557195
[pressedA] 1417606557290 [releasedA] 1417606557388

EMAIL[elenalg_93athotmail.com]:
[pressedE] 1417606560561 [releasedE] 1417606560628
[pressedL] 1417606560660 [releasedL] 1417606560762
[pressedE] 1417606560756 [releasedE] 1417606560857
[pressedN] 1417606560860 [releasedN] 1417606560955
[pressedA] 1417606561018 [releasedA] 1417606561117
[pressedL] 1417606561148 [releasedL] 1417606561215
[pressedG] 1417606561436 [releasedG] 1417606561535
```

Figura 2.2: Fragmento de un archivo generado por la aplicación.

2.1.2. Base de datos CMU.

La base de datos CMU es una base de datos pública desarrollada por Killourhy y Maxion [2] formada por 51 usuarios mecanógrafos de una comunidad universitaria, 30 hombres y 21 mujeres.
Cada usuario tecleó una contraseña que le presentaba el software, la misma para todos los usuarios, 50 veces en cada sesión. Si tecleaban mal la contraseña, debían repetirla. En total cada usuario realizó 8 sesiones, separadas temporalmente al menos

24 horas, por lo que cada usuario tecleó la contraseña 400 veces. La contraseña elegida fue *.tie5Roanl* por ser una contraseña robusta de 10 caracteres.

2.2. Algoritmo de reconocimiento.

2.2.1. Extracción de características.

La dinámica de tecleo se define mediante dos eventos, la presión de una tecla t^p y la liberación o alzado de tecla t^a. Sin embargo, las características más comúnmente empleadas y las utilizadas en este trabajo son combinaciones de ambos eventos, ver figura 2.3.
Estas características son:

- **Tiempo de espera (o Hold Time)**: es la diferencia temporal entre el instante en que se presiona una tecla i-ésima y el instante en que es liberada.

$$\mathrm{T}_i = t_i^a - t_i^p \quad i = 1, \ldots, N$$

- **Latencia Alzado-Presión (Release-Press Latency)**: es la diferencia temporal entre la presión de la $(i+1)$-ésima tecla y la liberación de la i-ésima.

$$L_i^{ap} = t_{i+1}^p - t_i^a \quad i = 1, \ldots, N-1$$

- **Latencia Presión-Presión (o Press-Press Latency)**: es la diferencia temporal entre la presión de la $(i+1)$-ésima tecla y la presión de la i-ésima.

$$L_i^{pp} = t_{i+1}^p - t_i^p \quad i = 1, \ldots, N-1$$

- **Latencia Alzado-Alzado (o Release-Release Latency)**: es la diferencia temporal entre la liberación de la $(i+1)$-ésima tecla y la liberación de la i-ésima.

$$L_i^{aa} = t_{i+1}^a - t_i^a \quad i = 1, \ldots, N-1$$

- **Latencia Presión-Alzado (o Press-Release Latency)**: es a diferencia temporal entre la liberación la $(i+1)$-ésima tecla y la presión de la i-ésima.

$$L_i^{pa} = t_{i+1}^a - t_i^p \quad i = 1, \ldots, N-1$$

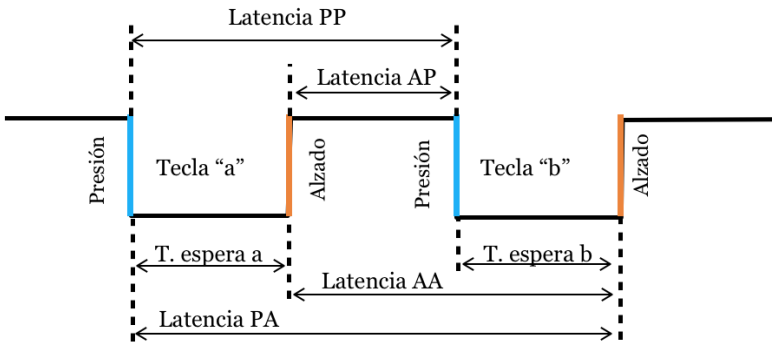

Figura 2.3: Características más comunes en la autenticación por dinámica de tecleo.

Como se ha visto en la sección 2.1.1, la aplicación extrae la información de tiempo absoluta en que cada tecla es presionada y liberada. A partir de esta información se extraen los vectores de las 5 características con las ecuaciones anteriores creando un archivo de texto para cada adquisición de la forma mostrada en la figura 2.4. La primera fila corresponde al vector de características del nombre, la segunda al de los apellidos, la tercera al del correo, la cuarta al de la nacionalidad y la quinta al del DNI. La existencia de valores negativos se debe a que un usuario presione una tecla sin haber liberado la anterior. Lejos de ser un problema, esto es un rasgo discriminatorio de la cadencia de escritura de ese usuario.

```
32 -36 95 94
126 126 126
31 31 -13 94 74 189 759 -60 221 220 886 158 189 63 157 95 63 31 -35 125 126 2 -35
97 202 95 2 126
157 189 315 157 62 157 63 315
```

Figura 2.4: Fichero de características.

2.2.2. Algoritmos de clasificación.

Se considera $\mathbf{f} = \{f_1, f_2, \ldots, f_M\}$ como el vector de M características de una muestra de test y $\mathbf{g^k} = \{g_1^k, g_2^k, \ldots, g_M^k\}$ $k \in 1, \ldots T$ como una adquisición con T muestras de entrenamiento. Los algoritmos que se emplean en este trabajo para la clasificación son:

- **Distancia Manhattan escalada:** basada en la propuesta por Araujo et al. [18]. La distancia entre un vector de características $\mathbf{f}$ de una muestra de test y

el conjunto de entrenamiento $\mathbf{g}$ se define como:

$$d_1 = ||\mathbf{f} - \bar{\mathbf{g}}||_1/\sigma$$

donde $||\cdot||_1$ es la distancia L_1, $\bar{\mathbf{g}}$ es el valor medio de las características del conjunto de entrenamiento y σ es el valor medio de la desviación típica de las características del conjunto de entrenamiento.

- **Distancia Mahalanobis combinada con búsqueda de vecinos más cercanos:** propuesta por Cho et al. [16]. La distancia entre un vector de características $\mathbf{f}$ de una muestra de test y cada uno de los vectores del conjunto de entrenamiento $\mathbf{g}^k$ se calcula como:

$$d_2^k = (\mathbf{f} - \mathbf{g^k})S^{-1}(\mathbf{f} - \mathbf{g^k})^T$$

donde se introduce la matriz de covarianza del conjunto de entrenamiento, S, para aumentar el impacto de aquellos vectores de características con menor varianza. La distancia final d_2 se obtiene como la menor distancia k calculada.

- **Distancia combinada Manhattan-Mahalanobis:** propuesta en [29]. Las muestras del vector de test $\mathbf{f}$ y el conjunto de entrenamiento $\mathbf{g}$ se normalizan inicialmente a través de la distancia Mahalanobis con $\hat{\mathbf{f}} = \phi\mathbf{f}$ y $\hat{\mathbf{g}} = \phi\mathbf{g}$, donde ϕ es la inversa de la raíz cuadrada de la matriz de covarianza del conjunto de entrenamiento. La distancia d_3 se calcula aplicando la distancia L_1 entre la muestra de test normalizada y la media del conjunto de entrenamiento normalizado:

$$d_3 = ||\hat{\mathbf{f}} - \bar{\hat{\mathbf{g}}}||_1$$

- **Distancia Manhattan escalada modificada:** se trata de una modificación de la distancia Manhattan escalada d_1. La distancia entre un vector de características $\mathbf{f}$ de una muestra de test y el conjunto de entrenamiento $\mathbf{g}$ se define como:

$$d_4 = ||\mathbf{f} - \bar{\mathbf{g}}||_1/\sigma'$$

donde σ' es la media de la desviación típica del conjunto de entrenamiento modificada:

$$\sigma_i' = \begin{cases} \frac{0{,}2}{M}\sum_{k=1}^{M}\sigma_k & si \;\; \sigma_i < \frac{0{,}2}{M}\sum_{k=1}^{M}\sigma_k \\ \sigma_i & resto \end{cases}$$

Esta modificación trata de mitigar los efectos de las muestras con poca varianza durante el proceso de normalización.

La elección de estos algoritmos se debe a su buen resultado documentado en la literatura [2] y a los buenos resultados experimentales obtenidos en este trabajo. En la tabla 2.2 se muestra que los 4 clasificadores elegidos son los que mejor rendimiento ofrecen con la base de datos CMU.

Tabla 2.2: Resumen del rendimiento de los mejores clasificadores con la base de datos CMU.

Clasificador	**EER (stdev)**
d1	0.096 (0.069)
d2	0.099 (0.064)
d3	0.084 (0.056)
d4	0.088 (0.062)
z-score [2]	0.102 (0.076)
SVM [2]	0.102 (0.065)
Mahalanobis [2]	0.110 (0.064)

2.3. Métodos de normalización de resultados del clasificador.

Existen diferentes formas de normalizar los resultados del clasificador. Analizados los resultados obtenidos en [22] para biometría de firma, en este trabajo se analizan tres técnicas de normalización divididas según la naturaleza de los datos utilizados. Tal que:

$$\hat{d} = \frac{d - \mu}{\sigma}$$

donde d representa los resultados del clasificador, $\hat{d}$ los resultados normalizados, y μ y σ representan la media y desviación típica de los resultados empleados para la normalización (d), los cuales variarán dependiendo del tipo de normalización empleada.

- **Normalización a posteriori basada en resultados genuinos e impostores (TIC, Target-Impostor Centric)**: se consideran las distribuciones de los resultados tanto genuinos como impostores. Se trata de una normalización a posteriori debido al uso del conjunto de datos de test. Se concatenan los resultados genuinos e impostores de un usuario y se calculan la media y la desviación estándar, que posteriormente se emplean para normalizar tanto los resultados genuinos como los impostores mediante la resta de la media y la división entre la desviación estándar.

- **Normalización basada en resultados genuinos (TC, Target Centric)**: se consideran únicamente los datos de entrenamiento del usuario genuino. Para calcular los estadísticos del conjunto de entrenamiento se aplica un sistema de validación cruzada dejando uno fuera (LOOCV, Leave-one-out cross-validation). Este método consiste en separar los datos de manera que en cada iteración se considera una muestra como muestra de test y el resto como muestras de entrenamiento. A partir de este método se obtiene la media y la desviación típica de los resultados de los datos de entrenamiento de cada usuario, que se emplean para normalizar tanto los resultados genuinos como los impostores del usuario correspondiente de igual manera que en la normalización anterior.

- **Normalización basada en resultados genuinos modificada**: se basa en la normalización anterior aplicando una modificación. Es lógico esperar que los resultados impostores tengan una mayor media y desviación estándar, por ello, antes de llevar a cabo la normalización, se aplica una modificación a los datos estadísticos del conjunto de entrenamiento dividiendo ambos por un valor experimental, lo que simula un comportamiento más real. Las modificaciones aplicadas a la media y desviación son las siguientes:

$$\mu' = \frac{\mu}{k}$$

siendo μ la media de los resultados empleados para la normalización, μ' la media modificada y k el ratio aplicado, que toma los valores 1.5, 3.2, 2 y 1.5, para la base de datos de desarrollo propio y 1.5, 1.5, 1.3 y 1.5, para la base de datos CMU, para $d1$, $d2$, $d3$ y $d4$ respectivamente.

$$\sigma' = \frac{\sigma}{m}$$

siendo σ la desviación típica de los resultados empleados para la normalización, σ' la desviación típica modificada y m el ratio aplicado, que toma los valores 0.75, 2.2, 1.2 y 0.7, para la base de datos de desarrollo propio y 1, 1, 0.8 y 1, para la base de datos CMU, para $d1$, $d2$, $d3$ y $d4$ respectivamente.

2.4. Medidas de rendimiento.

Las medidas empleadas para evaluar el rendimiento de características, clasificadores y normalizaciones son las siguientes:

- **Tasa de Falsa Aceptación (FAR)**: indica el porcentaje de usuarios impos-

tores que el sistema acepta incorrectamente como usuarios genuinos. Esto se debe a que, aunque el usuario sea impostor, si el resultado obtenido por el clasificador es mayor que el umbral de decisión es considerado genuino.

- **Tasa de Falso Rechazo (FRR)**: indica el porcentaje de usuarios genuinos considerados por el sistema incorrectamente como impostores. Esto sucede cuando el sistema falla al realizar una comparación entre el patrón de entrada y la plantilla de patrones de la base de datos. Por tanto, aunque un usuario sea genuino, si su resultado a la salida del clasificador, es menor que el umbral de decisión es considerado como impostor.

- **Tasa de Igual Error (EER)**: indica la tasa a la cual ambas tasas mencionadas anteriormente, FAR y FRR, son iguales, ver figura 2.5. La tasa de igual error es una medida muy comúnmente utilizada para medir el rendimiento de un sistema de una manera general. A menor EER, mejor es el rendimiento del sistema. Normalmente, en autenticación por dinámica de tecleo, se suele calcular un EER y un umbral de decisión para cada usuario debido a la gran variabilidad de los datos.

- **Curva Característica Operativa del Receptor (ROC)**: aporta una caracterización gráfica de la interrelación entre las tasas FAR y FRR, en una escala logarítmica. El eje de abscisas representa la tasa FAR, y el eje de ordenadas representa 1-FRR. Cuánto más próxima esté la curva ROC de la esquina superior izquierda de la representación, mejor será el rendimiento del sistema. Se muestra un ejemplo de una curva ROC en la figura2.6.

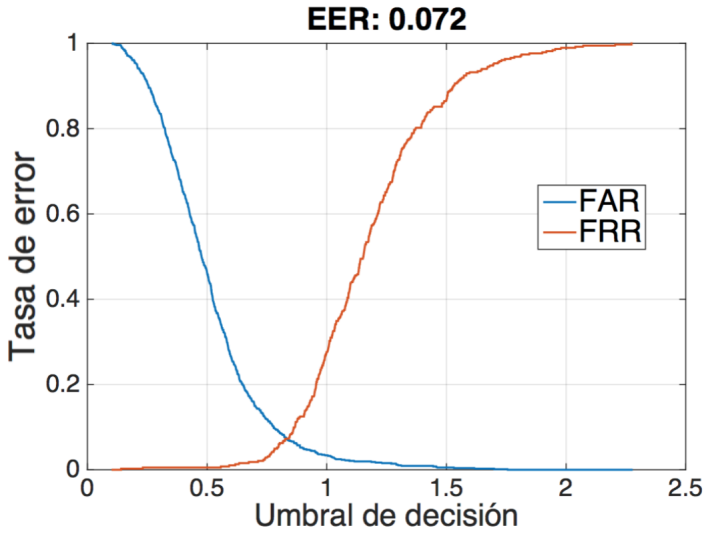

Figura 2.5: Representación curvas FAR y FRR. El punto en que se igualan determina la tasa EER.

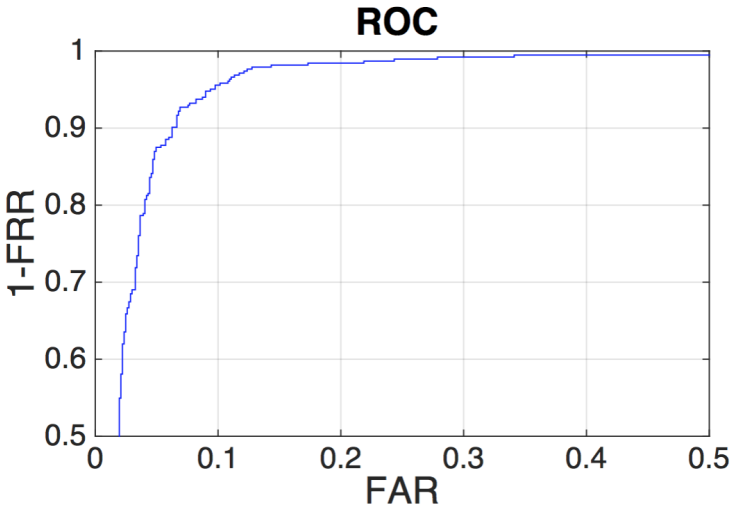

Figura 2.6: Curva ROC.

Capítulo 3

Experimentos.

3.1. Protocolo de experimentación.

3.1.1. Protocolo de experimentación con la base de datos de desarrollo propio.

Para cada usuario, se consideran dos conjuntos de muestras, el conjunto de entrenamiento y el conjunto de test. El conjunto de entrenamiento está formado por las 6 adquisiciones genuinas de la primera sesión, ver sección 2.1.1, y nos permite caracterizar a cada usuario. El conjunto de test está formado por las 6 adquisiciones genuinas de la segunda sesión y las 12 adquisiciones impostoras. Este conjunto se emplea para evaluar el rendimiento del sistema, las adquisiciones de test genuinas nos permiten calcular la tasa de falso rechazo, mientras que las adquisiciones de test impostoras nos permiten calcular la tasa de falta aceptación, dichas tasas se explican en la sección 2.4. Consideramos los datos personales introducidos por el usuario (nombre, apellidos, correo, nacionalidad y DNI) como 5 *passwords*. En el trabajo previo [28], se evalúan las capacidades discriminatorias de cada *password* de manera independiente y sus posibles combinaciones llegando a la conclusión de que el mejor rendimiento se obtiene con la combinación de los 5 datos. Por ello en este trabajo se considerarán los 5 *passwords* de manera conjunta.

Para calcular los resultados de las comparaciones genuinas y posteriormente calcular la tasa de falso rechazo, se consideran como conjunto de test las 6 adquisiciones genuinas de la segunda sesión. Se calcula la media de las 5 distancias obtenidas para cada password. Este valor se utilizará como resultado del clasificador (se puede ver como una combinación a nivel de resultados). Se obtienen por tanto 4 resultados o distancias de clasificación, una por cada clasificador visto en la sección 2.2.2. De esta manera se obtienen 5 distancias, una por cada *password*, de 4 clasificadores. Final-

mente, se calcula la media de las 5 distancias de cada clasificador y se obtiene una distancia final por cada muestra de test para cada clasificador. Se obtienen, por tanto, 64 (usuarios) x 6 (adquisiciones de test genuinas) = 384 resultados de comparaciones genuinas para cada uno de los 4 clasificadores.

El cálculo de los resultados de las comparaciones impostoras se realiza de manera similar al anterior, con la diferencia de que en este caso se considera como conjunto de test las 12 adquisiciones impostoras de cada usuario. Se obtienen, por tanto, 64 (usuarios) x 12 (adquisiciones de test impostoras) = 768 resultados de comparaciones impostoras para cada uno de los 4 clasificadores.

3.1.2. Protocolo de experimentación con la base de datos CMU.

De igual manera que con la anterior base de datos, para cada usuario se consideran dos conjuntos de datos, un conjunto de entrenamiento y un conjunto de test. El conjunto de entrenamiento lo constituirán las 50 muestras de las sesiones 2 a 4, ya que al tratarse de una contraseña desconocida para los usuarios, sección 2.1.2, el hecho de incluir la primera sesión empeora los resultados obtenidos. Respecto al conjunto de test, se emplea el protocolo sugerido en [2], detallado a continuación.

Para calcular los resultados de las comparaciones genuinas, que nos permitan calcular la tasa de falso rechazo, se consideran como conjunto de test las 50 muestras de las sesiones 5 a 8. Para cada usuario, se compara cada muestras de cada sesión de test con todas las muestras del conjunto de entrenamiento, con cada uno de los 4 clasificadores. Se obtienen así 51 (usuarios) x 4 (sesiones de test) x 50 (muestras/sesión) = 10.200 resultados de comparaciones genuinas por cada clasificador.

Para el cálculo de los resultados de las comparaciones impostoras, cada usuario es "atacado" por los otros 50 (debido a que todos introducen el mismo *password*). Se consideran como conjunto de test las 5 primeras muestras de la primera sesión de cada usuario atacante. Se realiza el mismo protocolo que para las comparaciones genuinas y se obtienen 51 (usuarios) x 50 (usuarios atacantes) x 1 (sesiones de test) x 5 (muestras/sesión) = 12.750 resultados de comparaciones impostoras por cada clasificador.

3.2. Resultados de los experimentos.

3.2.1. Resultados de los experimentos con la base de datos de desarrollo propio.

3.2.1.1. EER por usuario.

Como se ha mencionado anteriormente, debido a la gran variabilidad de los datos en un rasgo biométrico de comportamiento, la elección de un umbral, EER, único para cada usuario representa un escenario ideal en cuanto a rendimiento que analizaremos a continuación.

Mediante el protocolo explicado en la sección 3.1.1, se calculan las tasas FAR y FRR para los resultados de cada usuario, y en consecuencia su EER. A la hora de evaluar el rendimiento de los diferentes clasificadores se considera el EER promedio de todos los usuarios y su desviación estándar, tabla 3.1.

Como se ha visto en la sección 2.2.1, se tienen para cada usuario 5 vectores de características: *hold time* (H), latencia Alzado-Presión (latencia AP), latencia Presión-Presión (latencia PP), latencia Alzado-Alzado (latencia AA) y latencia Presión-Alzado (latencia PA). Tras probar múltiples combinaciones posibles, se ha observado que los mejores resultados se obtienen con la combinación de las características H + latencia AP. Como se puede apreciar en la tabla 3.1, el mejor rendimiento se consigue con el clasificador basado en la distancia Manhattan escalada modificada con dicha combinación de características.

El problema de este enfoque es que la elección del umbral óptimo para cada usuario a posteriori (usando resultados obtenidos con las muestras de test) es demasiado optimista pensando en una aplicación real. En un escenario real este cálculo no sería factible por lo que una solución más realista es calcular un único umbral para toda la base de datos.

3.2.1.2. EER único para toda la base de datos.

Como se menciona anteriormente, se trata de simular un escenario más real con un único umbral (EER) para toda la base de datos. Para ello, se concatenan todos los resultados de las comparaciones de todos los usuarios, y estos datos son los que se emplean para calcular una única FAR, FRR y EER. En la tabla 3.2 se muestran los resultados obtenidos para los diferentes clasificadores y características. De nuevo, los mejores resultados se observan en la combinación de características *hold time* + latencia Alzado-Presión para el clasificador basado en distancia Manhattan escalada modificada. Se observa como las tasas de error se incrementan considerablemente,

Table 3.1: Rendimiento de los clasificadores expresado en términos de EER promedio y desviación estándar para el caso de un EER único por usuario. En negrita se muestra el mejor rendimiento.

	Manhattan escalada		**Mahalanobis + KNN**		**Manhattan + Mahalanobis**		**Manhattan escalada modificada**	
	EER	**stdev**	**EER**	**stdev**	**EER**	**stdev**	**EER**	**stdev**
H	0.111	0.116	0.159	0.130	0.138	0.118	0.088	0.119
PA	0.062	0.095	0.121	0.127	0.128	0.118	0.046	0.822
AA	0.073	0.109	0.105	0.135	0.102	0.113	0.053	0.094
PP	0.067	0.100	0.119	0.115	0.121	0.119	0.044	0.084
AP	0.070	0.100	0.119	0.124	0.104	0.122	0.041	0.085
H+AP	0.043	0.076	0.090	0.116	0.096	0.109	**0.022**	**0.063**

Table 3.2: Rendimiento de los clasificadores expresado en términos de EER para el caso de un EER único para la base de datos. En negrita se muestra el mejor rendimiento.

	Manhattan escalada	**Mahalanobis + KNN**	**Manhattan + Mahalanobis**	**Manhattan escalada modificada**
	EER	**EER**	**EER**	**EER**
H	0.177	0.229	0.206	0.148
PA	0.127	0.169	0.174	0.088
AA	0.129	0.143	0.177	0.088
PP	0.125	0.176	0.185	0.081
AP	0.139	0.158	0.154	0.085
H+AP	0.120	0.143	0.164	**0.073**

llegando incluso a duplicarse en algunos casos. Esto nos hace pensar que los resultados obtenidos por los usuarios no están alineados y el uso de un umbral común para todos tiene efectos muy claros en el rendimiento.

En la figura 3.1 se muestran gráficas comparativas de las curvas ROC, ver sección 2.4, de los 4 clasificadores para cada característica. Se observa que en todos los casos, al igual que en el experimento anterior, la distancia Manhattan escalada modificada es la que ofrece mejores resultados.

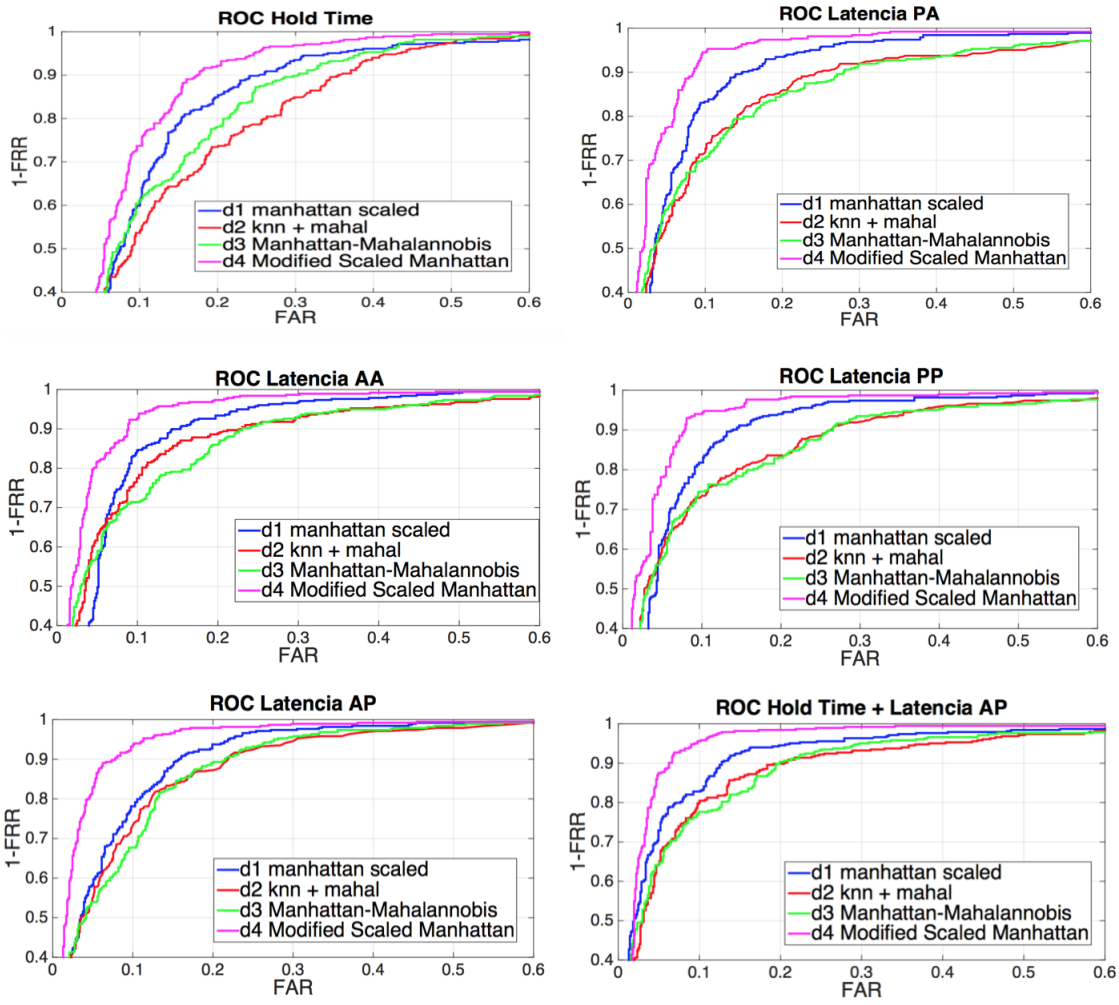

Figura 3.1: Evaluación mediante curvas ROC de las diferentes características con los 4 clasificadores.

3.2.1.3. EER único para toda la base de datos con normalización a posteriori basada en resultados genuinos e impostores (TIC, Target-Impostor Centric).

El primer método de normalización que se evalúa es la normalización a posteriori basada en resultados genuinos e impostores, explicada en la sección 2.3. En la tabla 3.3 se muestra una evaluación de esta normalización para cada característica y clasificador. De nuevo, los mejores resultados se consiguen con el clasificador basado en distancia Manhattan escalada modificada. A pesar de que se consigue una tasa de error (EER) muy reducida, 3.1 %, esta normalización no es la más conveniente en un escenario realista. Se trata de una normalización ideal ya que se hace uso de datos del conjunto de test, los cuales en una aplicación real son desconocidos. Estos resultados nos hacen pensar que una correcta normalización podría ayudar a mitigar los efectos de resultados no alineados por cada usuario.

Table 3.3: Rendimiento de los clasificadores expresado en términos de EER para el caso de normalización a posteriori. En negrita se muestra el mejor rendimiento.

	Manhattan escalada	**Mahalanobis + KNN**	**Manhattan + Mahalanobis**	**Manhattan escalada modificada**
	EER	**EER**	**EER**	**EER**
H	0.120	0.191	0.160	0.092
PA	0.082	0.146	0.136	0.050
AA	0.084	0.113	0.125	0.050
PP	0.072	0.137	0.134	0.047
AP	0.070	0.129	0.135	0.044
H+AP	0.050	0.105	0.100	**0.031**

3.2.1.4. EER único para toda la base de datos con normalización basada en resultados genuinos (TC, Target Centric).

En una aplicación real la normalización se suele llevar a cabo en la fase de entrenamiento, sección 1.2, con los datos de los usuarios genuinos. Este experimento trata de emular ese comportamiento, el procedimiento empleado se explica en la sección 2.3. En la tabla 3.4 se muestra una evaluación del rendimiento de esta normalización para cada característica y clasificador. Los resultados empeoran respecto a la normalización anterior un factor aproximado de 2. Creemos que este empeoramiento se debe a no considerar en la normalización el efecto de los resultados impostores, ya que estos son fundamentales para obtener la tasa de falsa aceptación (FAR) y en

consecuencia su impacto en el rendimiento final es muy elevado. El mejor resultado sigue siendo, con mucha diferencia, el clasificador basado en distancia Manhattan escalada modificada.

Table 3.4: Rendimiento de los clasificadores expresado en términos de EER para el caso de normalización a priori. En negrita se muestra el mejor rendimiento.

	Manhattan escalada	**Mahalanobis + KNN**	**Manhattan + Mahalanobis**	**Manhattan escalada modificada**
	EER	**EER**	**EER**	**EER**
H	0.200	0.282	0.269	0.1642
PA	0.112	0.299	0.201	0.094
AA	0.182	0.279	0.209	0.092
PP	0.104	0.283	0.185	0.089
AP	0.116	0.271	0.191	0.091
H+AP	0.106	0.242	0.172	**0.075**

3.2.1.5. EER único para toda la base de datos con normalización basada en resultados genuinos modificada.

Esta modificación, explicada en la sección 2.3, tiene por objetivo mejorar la normalización introduciendo el efecto de los resultados impostores. Los resultados obtenidos, mostrados en la tabla 3.5, en efecto, mejoran respecto a la misma normalización sin la modificación. En el caso del clasificador basado en distancia Manhattan escalada modificada la tasa de error (EER) disminuye de un 7.5 % a un 5.5 %, no sólo es el clasificador que mejor resultado proporciona, sino que además es el que más porcentaje de mejora experimenta con esta modificación.

Table 3.5: Rendimiento de los clasificadores expresado en términos de EER para el caso de normalización a priori modificada. En negrita se muestra el mejor rendimiento.

	Manhattan escalada	**Mahalanobis + KNN**	**Manhattan + Mahalanobis**	**Manhattan escalada modificada**
	EER	**EER**	**EER**	**EER**
H	0.229	0.297	0.273	0.167
PA	0.130	0.173	0.182	0.078
AA	0.201	0.161	0.193	0.104
PP	0.119	0.153	0.166	0.077
AP	0.131	0.147	0.171	0.078
H+AP	0.096	0.156	0.156	**0.055**

3.2.1.6. Discusión de los resultados obtenidos.

En vista de los resultados observados, cabe destacar que la combinación de características más discriminatoria es la formada por el tiempo de espera, o *hold time*, y la latencia Alzado-Presión. En todos los experimentos esta combinación presenta la menor tasa de igual error, EER. Por otro lado el clasificador que ofrece los mejores resultados es el basado en la distancia Manhattan escalada modificada. Esto era de esperar ya que en un rasgo como la dinámica de tecleo, no es difícil que un usuario presente una cadencia de tecleo puntual muy definida y constante, lo que haría que adquisiciones muy similares, si no iguales, resultaran en puntuaciones muy por encima de la media empeorando el resultado global. La modificación aplicada mitiga estos efectos mejorando notablemente el resultado.

En la figura 3.2 se muestra una comparativa de las curvas ROC con la combinación de características óptima y el clasificador óptimo para los casos analizados.

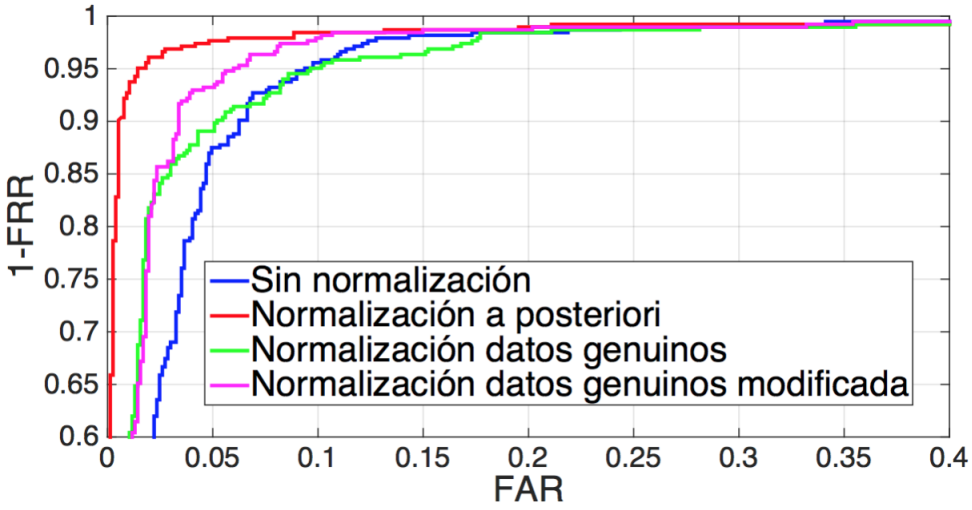

Figura 3.2: Comparativa curvas ROC para la combinación de características Hold time + Latencia Alzado-Presión y el clasificador basado en distancia Manhattan modificada.

A la derecha de la figura 3.3 se muestra el escenario de rendimiento óptimo en el cual se tiene un umbral óptimo único para cada usuario, pero como hemos dicho anteriormente, este escenario no es muy realista. Nuestro punto de partida es un escenario sin ningún tipo de normalización en el que se obtiene un EER = 0.073. El caso ideal óptimo de normalización es el caso de la normalización a posteriori en el que se obtiene un EER = 0.031, lo que supone una mejora del 57.54 %, sin embargo, como se explica anteriormente, no se trata de un escenario real. Un caso más realista

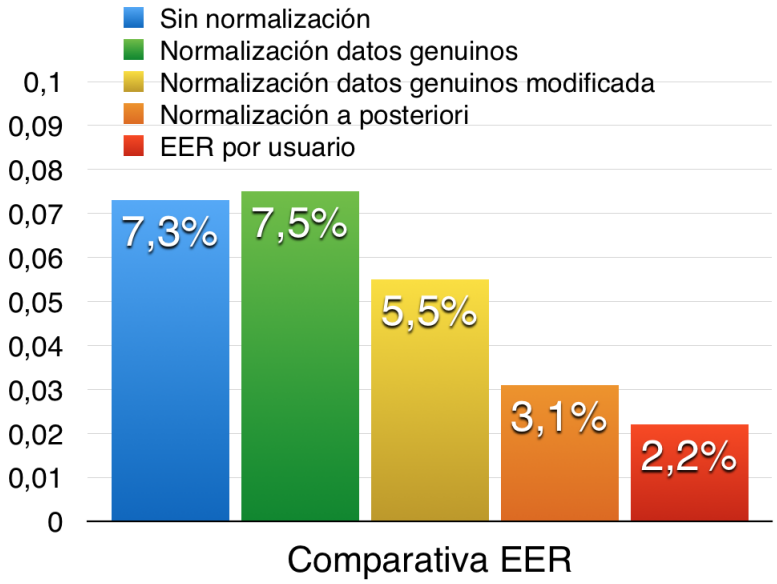

Figura 3.3: Comparativa EER para la combinación de características Hold time + Latencia Alzado-Presión y el clasificador basado en distancia Manhattan modificada.

es el de la normalización a priori, con el cual se obtiene un EER = 0.075. En este caso el rendimiento se mantiene prácticamente constante sin mejora alguna, esto se debe a que las distribuciones de los impostores tienen un impacto importante que se debe tener en cuenta. Para tratar de simular el impacto de los impostores se realiza la modificación explicada anteriormente en la cual se modifican la media y la desviación estándar empleadas para la normalización con el fin de, como hemos dicho, simular este comportamiento. Con esta modificación se obtiene un EER = 0.055, el cual supone una mejora del 24.66 % respecto al punto de partida. Estos datos se muestran gráficamente en la figura 3.3.

3.2.2. Resultados de los experimentos con la base de datos CMU.

A diferencia de la base de datos de desarrollo propio, la base de datos CMU presenta un único vector de características para cada usuario. Para construir el vector de características consideraron la pulsación de la tecla Intro como parte de la contraseña, teniendo entonces una contraseña de longitud 11. Se consideraron la latencia Presión-Presión, la latencia Alzado-Presión y el *hold time* [2], teniendo entonces un vector de características para cada usuario de longitud 31.

3.2.2.1. EER por usuario.

Mediante el protocolo explicado en la sección 3.1.2, se calculan las tasas FAR, FRR y EER para cada uno de los 51 usuarios. A la hora de evaluar el rendimiento de los diferentes clasificadores se considera el EER promedio de todos los usuarios y su desviación estándar, los resultados se muestran en la tabla 3.6. Se observa que los resultados son muy similares para los 4 clasificadores sin apenas variación. Esto se puede deber a que el tener muchas muestras genera un comportamiento similar en los 4 clasificadores.

Table 3.6: Rendimiento de los clasificadores expresado en términos de EER promedio y desviación estándar para el caso de un EER único por usuario para la base de datos CMU.

Manhattan escalada		**Mahalanobis + KNN**		**Manhattan + Mahalanobis**		**Manhattan escalada modificada**	
EER	**stdev**	**EER**	**stdev**	**EER**	**stdev**	**EER**	**stdev**
0.086	0.059	0.089	0.049	0.082	0.054	0.088	0.063

3.2.2.2. EER único para toda la base de datos.

Se sigue el mismo protocolo que en el experimento anterior para obtener los resultados individuales, pero estos se concatenan de manera que se obtiene una FAR, FRR y EER para todos los usuarios. En la tabla 3.7 se muestran los umbrales obtenidos para cada clasificador. De nuevo no se aprecia gran diferencia entre ellos, aunque el clasificador basado en la distancia Manhattan escalada modificada se distancia algo más del resto. Se observa que el grado de deterioro es menor en esta base de datos. Esto nos hace pensar que los resultados de los diferentes usuarios podrían estar más alineados que en la base de datos de desarrollo propio.

Table 3.7: Rendimiento de los clasificadores expresado en términos de EER promedio para el caso de un EER único para toda la base de datos CMU.

Manhattan escalada	**Mahalanobis + KNN**	**Manhattan + Mahalanobis**	**Manhattan escalada modificada**
EER	**EER**	**EER**	**EER**
0.103	0.096	0.099	0.126

3.2.2.3. EER único para toda la base de datos con normalización a posteriori basada en resultados genuinos e impostores (TIC, Target-Impostor Centric).

En este experimento se aplica la normalización a posteriori explicada en la sección 2.3, y del mismo modo que en el experimento anterior se calcula una FAR, FRR y EER para toda la base de datos. Los resultados obtenidos se muestran en la tabla 3.8. Esta normalización nos proporciona una mínima mejora respecto al caso anterior sin normalización. La normalización a posteriori nos permite obtener resultados muy próximos a los ideales obtenidos con un EER por usuario.

Table 3.8: Rendimiento de los clasificadores expresado en términos de EER promedio para el caso de normalización a posteriori de la base de datos CMU.

Manhattan escalada	**Mahalanobis + KNN**	**Manhattan + Mahalanobis**	**Manhattan escalada modificada**
EER	**EER**	**EER**	**EER**
0.086	0.090	0.079	0.089

3.2.2.4. EER único para toda la base de datos con normalización basada en resultados genuinos (TC, Target Centric).

En este experimento se aplica la normalización basada en resultados genuinos, sección 2.3, mediante la cual se obtiene un EER único para cada clasificador, mostrados en la tabla 3.9. Los resultados se siguen manteniendo prácticamente iguales para los 4 clasificadores, empeorando todos mínimamente. Al igual que con la base de datos propia, la normalización basada en datos genuinos no produce mejoras. Es razonable pensar que los resultados impostores tienen un impacto que es necesario considerar.

Table 3.9: Rendimiento de los clasificadores expresado en términos de EER para el caso de normalización a priori con la base de datos CMU.

Manhattan escalada	**Mahalanobis + KNN**	**Manhattan + Mahalanobis**	**Manhattan escalada modificada**
EER	**EER**	**EER**	**EER**
0.107	0.115	0.010	0.117

3.2.2.5. EER único para toda la base de datos con normalización basada en resultados genuinos modificada.

A diferencia de nuestra base de datos, en la cual la modificación mejora considerablemente los resultados, en este caso esto no ocurre debido a que el impacto de los impostores no es tan relevante, ya que todos actúan como impostores. En la tabla 3.10, en la que se muestran los resultados, se observa que la modificación no mejora los resultados, los cuales se mantienen similares para los 4 clasificadores. En este caso, la normalización propuesta no tiene ningún efecto positivo en los datos. No se consiguen mejorar los resultados lo suficiente como para considerar una mejora significativa.

Table 3.10: Rendimiento de los clasificadores expresado en términos de EER para el caso de normalización a priori modificada con la base de datos CMU.

Manhattan escalada	**Mahalanobis + KNN**	**Manhattan + Mahalanobis**	**Manhattan escalada modificada**
EER	**EER**	**EER**	**EER**
0.102	0.105	0.098	0.107

3.2.2.6. Discusión de los resultados obtenidos.

En la figura 3.4 se muestra una comparativa del rendimiento de cada clasificador en cada experimento analizado. Se observa que en los clasificadores basado en distancia Manhattan escalada y distancia Manhattan + Mahalanobis la normalización a priori no produce ningún efecto, sin embargo los clasificadores basados en distancia Mahalanobis + KNN y distancia Manhattan escalada modificada son más sensibles a la normalización, la cual empeora los resultados en el primero y los mejora en el segundo. La razón de que no se produzca ninguna mejora con las normalizaciones propuestas creemos puede estar influenciada por el hecho de usar un mismo *password* para todos los usuarios. Esto, unido a la gran cantidad de repeticiones de los experimentos (cada usuario repitió el *password* al menos 400 veces) hace que los resultados se encuentren más alineados que en el caso de la base de datos propia.

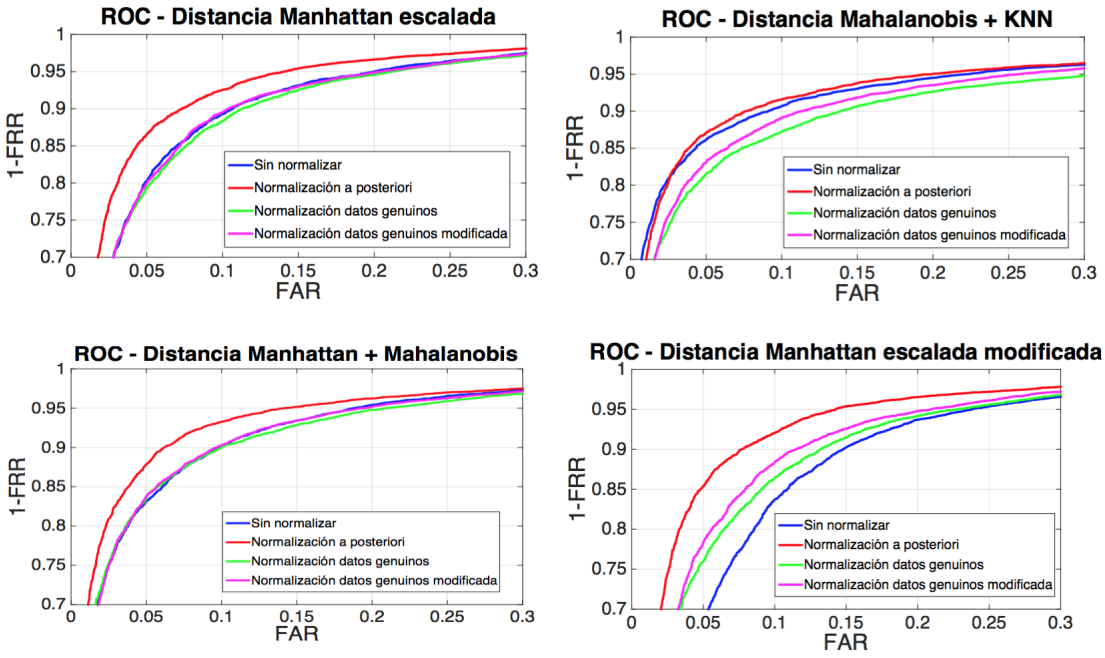

Figura 3.4: Comparativa curvas ROC para cada clasificador con la base de datos CMU.

Capítulo 4

Conclusiones y trabajo futuro.

4.1. Conclusiones.

Este trabajo se marca como principal objetivo el estudio del impacto de la normalización de resultados en un sistema biométrico basado en la dinámica de tecleo. Para ello se ha participado en la adquisición de una base de datos de desarrollo propio y en el testeo de la aplicación empleada para ello. Mediante la realización de diversos experimentos (explicados en el capítulo 3) se ha estudiado el impacto de la normalización en la base de datos de desarrollada y en la base de datos pública CMU.

En la base de datos de desarrollo propio se han realizado experimentos con las diferentes características disponibles (*hold time,* latencia Presión-Alzado, latencia Alzado-Presión, latencia Alzado-Alzado, latencia Presión-Presión y latencia Alzado-Presión) y posibles combinaciones llegando a la conclusión de que la mejor combinación es la formada por las características *hold time* y latencia Alzado-Presión. A su vez, se ha analizado también el rendimiento de 4 clasificadores basados en distancias: Manhattan escalada, Mahalanobis combinada con búsqueda de vecinos más cercanos, combinada Manhattan-Mahalanobis y Manhattan escalada modificada. En todos los experimentos realizados el clasificador que presentaba mejor rendimiento es el basado en distancia Manhattan escalada modificada, proporcionando siempre una mejora considerable respecto al resto de clasificadores. Esta mejora se debe a la mitigación que produce la modificación empleada sobre la distancia Manhattan escalada de los efectos de las muestras con reducida varianza. Es decir, adquisiciones, normalmente genuinas de un mismo usuario, con características muy similares o iguales producen un efecto de sobrepuntuación que empeora los resultados, la modificación propuesta mitiga estos efectos mejorando el rendimiento. En el análisis del impacto de la normalización de resultados se han estudiado diferentes casos, se han

considerado para los resultados la combinación de características y el clasificador de mejor rendimiento. El caso ideal en cuanto a rendimiento en un sistema basado en un rasgo de comportamiento consiste en la elección de un umbral óptimo por usuario, en este caso se obtiene un EER = 2.2 %. Sin embargo, en un escenario más realista con un único umbral para la base de datos obtiene un EER = 7.3 %. Si se aplica al caso anterior una normalización a posteriori, es decir, empleando los resultados tanto genuinos como impostores obtenemos un EER = 3.1 %. Esta normalización mejora un 57.54 % respecto al mismo experimento sin normalización, pero la normalización a posteriori es difícilmente aplicable en un escenario real de autenticación. Por ello se lleva a cabo la normalización a priori empleando únicamente datos genuinos obteniendo un EER = 7.5 %. Este enfoque no presenta ninguna mejora significativa debido a que no se tienen en cuenta los resultados de los usuarios impostores, los cuales son fundamentales en el cálculo de las curvas FAR y en el rendimiento final. Para solventar este problema se propone una modificación de dicha normalización de manera que se tengan en cuenta los resultados impostores, sin necesidad de emplear muestras impostoras, se obtiene EER = 5.5 %, que supone una mejora del 24.66 %.

En cuanto a la base de datos CMU, tras realizar experimentos similares a los relatados anteriormente, se observa que ningún clasificador destaca por encima de los demás, teniendo todos un rendimiento similar, lo que se puede deber al elevado número de muestras y por tanto un mejor modelado de cada usuario. Se ha observado también que la normalización de resultados, a diferencia de la otra base de datos, no aporta ninguna mejora significativa al rendimiento del sistema. Esta diferencia de comportamiento entre las dos bases de datos, creemos se debe a la diferente naturaleza de las mismas. El hecho de que en la base de datos CMU todos los usuarios empleen el mismo *password* y lo tecleen muchas veces provoca que los resultados se encuentren más alineados y que el impacto de los resultados impostores no sea tan relevante, por ello no se producen mejoras significativas.

4.2. Trabajo futuro.

Como trabajo futuro se proponen los siguientes objetivos:

- Aumentar la base de datos desarrollada para obtener resultados más fiables y precisos.

- Estudiar la normalización en diferentes bases de datos para ver como se adapta a los diferentes escenarios de aplicación.

- Estudiar el impacto de la normalización en autenticación continua, es decir, en reconocimiento basado en dinámica de tecleo en texto libre.

- Probar diferentes algoritmos de reconocimiento, como pueden ser las redes neuronales, las máquinas de soporte vectorial, etc.

- Estudio de otras técnicas de normalización que traten de mejorar los resultados obtenidos.

4.3. Contribución en congreso internacional.

Los resultados de este trabajo se han documentado en un artículo científico que ha sido aceptado en la conferencia internacional *49th Annual International Carnahan Conference on Security Technology* (ICCST), la cual tendrá lugar del 21-24 de septiembre de 2015 en Taipei, Taiwan. Se adjuntan el *abstract* aceptado y el certificado de aceptación.

Score Normalization for Keystroke Dynamics Biometrics

Elena Luna-Garcia, Aythami Morales, Julian Fierrez, Javier Ortega-Garcia

Departamento de Tecnología Electrónica y de las Comunicaciones, EPS, Universidad Autonoma de Madrid, C\ Francisco Tomás y Valiente, 11, 28049 Madrid, Spain
elenalunagarcia@gmail.com, aythami.morales@uam.es, julian.fierrez@uam.es, javier.ortega@uam.es

Abstract—**Among all the biometric technologies, keystroke dynamic recognition is especially interesting for the Cyber Security because of: i) no need of extra sensors as the recognition of users is done according to their typing patterns using a keyboard or keypad; ii) it is possible to realize a continuous authentication based on the monitoring of the user behavior; iii) keystroke dynamic technologies can be easily integrated in web-platforms or web-services.**

The flowchart of typical keystroke dynamic recognition systems includes a classification phase in which query samples are compared with a stored template. The identity of the user will be authenticated if the distance between the template and the query sample is lower than a pre-defined threshold. How to define this threshold is a challenge that has to be addressed before the deployment of a biometric system in real operational environments. For the best of our knowledge, this topic has been scarcely analyzed by the research community for the case of keystroke dynamics biometric systems.

Previous studies have shown that the performance of behavioral biometric recognition systems (e.g. voice and signature) can be largely improved with score normalization and target-dependent techniques. The main objective of this work is twofold: i) to analyze the effects of different thresholding techniques in 4 different keystroke dynamics recognition systems for real operational scenarios; ii) to improve the performance of keystroke dynamics on the basis of target-dependent score normalization techniques. The experiments included in this work are worked out over the keystroke pattern of 100 users. The experiments show that there is large room for improvements in keystroke dynamic systems. The results suggest that target-dependent sore normalization techniques can be used to improve the performance of keystroke dynamics systems in more than 20%. These results encourage researchers to explore this research line to further improve the performance of these systems in real operational environments.

Topic: Biometrics including voice, hand, finger, face and other characteristics

Keywords — keystroke dynamics, biometrics, score normalization.

Dear Aythami Morales, Julian Fierrez, Javier Ortega-Garcia and Elena Luna,

We are very pleased to inform you that your paper entitled "Score Normalization for Keystroke Dynamics Biometrics", has been ACCEPTED for presentation in the 49th Annual International Carnahan Conference on Security Technology (ICCST), which is scheduled to be held in Taipei, Taiwan during the period 21 – 24 September 2015.

Please submit your full paper to Ms. Min-Chen Su at ncu57976@cc.ncu.edu.tw or mcsumingchuansu@gmail.com and confirm your registration by 26th June 2015. The papers that are accepted must be presented at the conference, either by the authors themselves, or by proxy. According to our No-show Policy, the papers that are not presented will be deleted from the final ICCST2015 proceedings and from IEEE Xplore.

If you are not the presenter of the paper, please forward this email to the designated presenter.

The information on paper submission and copyright transfer to IEEE and others, please refer to ICCST 2015 official website (http://www.iccst2015.org/) for the details.

We look forward to seeing you at the conference.

Sincerely

朱延祥 朱延祥 *(Yen-Hsyang Chu)*

Chair of Local Organization Committee for the ICCST2015
Add : Research Center for Advanced Science and Technology , National Central Univ.
No. 300, Jung-Da Road, Jungli City, Taoyuan, Taiwan 32001, R.O.C.
E-mail：yhchu@jupiter.ss.ncu.edu.tw

Bibliografía

[1] R. Tolosana *et al.*, "Estudio de interoperabilidad en sistemas biométricos de firma manuscrita dinámica," 2014. Universidad Autónoma de Madrid.

[2] K. S. Killourhy and R. A. Maxion, "Comparing anomaly-detection algorithms for keystroke dynamics," in *Dependable Systems & Networks, 2009. DSN'09. IEEE/IFIP International Conference on*, pp. 125–134, IEEE, 2009.

[3] A. K. Jain, P. Flynn, and A. A. Ross, *Handbook of biometrics*. Springer Science & Business Media, 2007.

[4] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 14, no. 1, pp. 4–20, 2004.

[5] A. K. Jain, A. A. Ross, and K. Nandakumar, *Introduction to biometrics*. Springer Science & Business Media, 2011.

[6] R. Joyce and G. Gupta, "Identity authentication based on keystroke latencies," *Communications of the ACM*, vol. 33, no. 2, pp. 168–176, 1990.

[7] H. Saevanee and P. Bhattarakosol, "Authenticating user using keystroke dynamics and finger pressure," in *Consumer Communications and Networking Conference, 2009. CCNC 2009. 6th IEEE*, pp. 1–2, IEEE, 2009.

[8] J. Roth, X. Liu, A. Ross, and D. Metaxas, "Investigating the discriminative power of keystroke sound," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 2, p. 333, 2015.

[9] F. Monrose and A. D. Rubin, "Keystroke dynamics as a biometric for authentication," *Future Generation computer systems*, vol. 16, no. 4, pp. 351–359, 2000.

[10] G. E. Forsen, M. R. Nelson, and R. J. Staron Jr, "Personal attributes authentication techniques.," tech. rep., DTIC Document, 1977.

[11] R. Spillane, "Keyboard apparatus for personal identification," *IBM Technical Disclosure Bulletin*, vol. 17, no. 3346, p. 3346, 1975.

[12] R. S. Gaines, W. Lisowski, S. J. Press, and N. Shapiro, "Authentication by keystroke timing: Some preliminary results," tech. rep., DTIC Document, 1980.

[13] F. Monrose and A. Rubin, "Authentication via keystroke dynamics," in *Proceedings of the 4th ACM conference on Computer and communications security*, pp. 48–56, ACM, 1997.

[14] C. C. Loy, W. K. Lai, and C. P. Lim, "Keystroke patterns classification using the artmap-fd neural network," in *Intelligent Information Hiding and Multimedia Signal Processing, 2007. IIHMSP 2007. Third International Conference on*, vol. 1, pp. 61–64, IEEE, 2007.

[15] Y. Deng and Y. Zhong, "Keystroke dynamics user authentication based on gaussian mixture model and deep belief nets," *International Scholarly Research Notices*, vol. 2013, 2013.

[16] S. Cho, C. Han, D. H. Han, and H.-I. Kim, "Web-based keystroke dynamics identity verification using neural network," *Journal of organizational computing and electronic commerce*, vol. 10, no. 4, pp. 295–307, 2000.

[17] G. A. Betancourt, "Las máquinas de soporte vectorial (svms)," *Scientia et Technica*, vol. 1, no. 27, 2005.

[18] L. C. Araújo, L. H. Sucupira Jr, M. G. Lizarraga, L. L. Ling, and J. B. T. Yabu-Uti, "User authentication through typing biometrics features," *Signal Processing, IEEE Transactions on*, vol. 53, no. 2, pp. 851–855, 2005.

[19] A. Morales, J. Fierrez, and J. Ortega-Garcia, "Towards predicting good users for biometric recognition based on keystroke dynamics," in *Computer Vision-ECCV 2014 Workshops*, pp. 711–724, Springer, 2014.

[20] A. Jain, K. Nandakumar, and A. Ross, "Score normalization in multimodal biometric systems," *Pattern recognition*, vol. 38, no. 12, pp. 2270–2285, 2005.

[21] J. Ortega-Garcia, J. Fiérrez-Aguilar, J. Martin-Rello, and J. Gonzalez-Rodriguez, "Complete signal modeling and score normalization for function-based dynamic signature verification," in *Audio-and Video-based Biometric Person Authentication*, pp. 658–667, Springer, 2003.

[22] J. Fierrez-Aguilar, J. Ortega-Garcia, and J. Gonzalez-Rodriguez, "Target dependent score normalization techniques and their application to signature verification," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 35, no. 3, pp. 418–425, 2005.

[23] T. Matsui, T. Nishitani, and S. Firui, "Robust methods of updating model and a priori threshold in speaker verification," in *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*, vol. 1, pp. 97–100, IEEE, 1996.

[24] A. K. Jain, F. D. Griess, and S. D. Connell, "On-line signature verification," *Pattern recognition*, vol. 35, no. 12, pp. 2963–2972, 2002.

[25] E. Vural, J. Huang, D. Hou, and S. Schuckers, "Shared research dataset to support development of keystroke authentication," in *Biometrics (IJCB), 2014 IEEE International Joint Conference on*, pp. 1–8, IEEE, 2014.

[26] R. Giot, M. El-Abed, and C. Rosenberger, "Greyc keystroke: a benchmark for keystroke dynamics biometric systems," in *Biometrics: Theory, Applications, and Systems, 2009. BTAS'09. IEEE 3rd International Conference on*, pp. 1–6, IEEE, 2009.

[27] Y. Li, B. Zhang, Y. Cao, S. Zhao, Y. Gao, and J. Liu, "Study on the beihang keystroke dynamics database," in *Biometrics (IJCB), 2011 International Joint Conference on*, pp. 1–5, IEEE, 2011.

[28] M. Falanga, "Web-based biometric recognition using keystroke dynamics," 2014. Universita' degli Studi di Napoli Federico II.

[29] Y. Zhong, Y. Deng, and A. K. Jain, "Keystroke dynamics for user authentication," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pp. 117–123, IEEE, 2012.